IN THE CLAIMS

Please amend the claims as follows:

1. (amend) Device for processing data, comprising:

a processor for executing program routines, and a memory for storing program routines to be executed by said processor, where

at least a part of said memory is arranged as a protected part from which data can be read but which is protected against being written into, where said protected part is arranged such that a mechanism is provided such that after data is initially stored in said protected part, any subsequent writing of data into said protected part is irreversibly blocked, and

said processor is arranged to necessarily execute a program routine stored in said protected part of said memory upon start-up.

2. (previously amended) Device according to claim 1, wherein said processor stores permanent start addresses that are necessarily called upon start-up of said processor, where at least one of said start addresses points to said protected part of said memory.

3. (previously amended) Device according to claim 1, wherein said protected part of said memory is a first part, and said memory further comprises a second part into which data can be written, where the program routine from said protected part executed

by said processor upon start-up comprises checking for changes in at least a part of the data contained in said second part.

4. (previously amended) Device according to claim 3, wherein said program routine from said protected part executed by said processor upon start-up comprises calculating a characteristic parameter for data being checked for changes, and comparing said characteristic parameter with a value stored in said second part of said memory at the time of writing said data being checked for changes into said second part of said memory.

5. (original) Device according to claim 4, wherein said characteristic parameter is a check sum.

6. (previously amended) Device according to claims 1, wherein said memory comprises a plurality of memory devices, one of which comprises said protected part, and the rest of which are arranged such that data may be written into them.

7. (cancel)

8. (previously amended) Device according to claim 7, wherein said protected area is arranged such that the process for storing data therein comprises:

writing data into said protected part via a write line, and

sending a signal to said protected part in response to which said write line is permanently interrupted.

C1

9. (original) Device according to claim 8, wherein said write line is a fusable link.

10. (previously amended) Device according to claim 1, wherein said memory comprises a finite state machine, said finite state machine defining a state which protects said protected part from being written into.

11. (previously amended) Device according to claim 1, wherein said memory comprise one or more of an EEPROM, a flash memory device, and a flash memory device emulating an EEPROM.

12. (previously amended) Device according to claim 1, wherein said memory comprises a memory chip having electrical contacts for being connected with a circuit board that are arranged such that said electrical contacts are covered by said memory chip when said memory chip is mounted on said circuit board.

13. (previously amended) Device according to claim 12, wherein said electrical contacts are provided in a ball-grid-array.

14. (previously amended) Communication device comprising a device for processing data according to claim 1.

15. (original) Communication device according to claim 14, wherein said communication device is a mobile telephone.

16. (original) Communication device according to claim 14, wherein said communication device is a bluetooth communication device.

17. (amended) Method for controlling a data processing device having a processor for executing program routines and a memory for storing program routines to be executed by said processor, comprising:

~~where~~ arranging at least a part of said memory ~~is arranged~~ as a protected part from which data can be read but which is protected against being written into;~~, comprising:~~

after data is initially stored in said protected part, irreversibly blocking any subsequent writing of data into said protected part; and

~~letting~~ said processor necessarily ~~execute~~ executing a program routine stored in said protected part of said memory upon start-up.

18. (original) Method according to claim 17, wherein said processor stores permanent start addresses that are necessarily called upon start-up of said processor, where at least one of said start addresses points to said protected part of said memory.

19. *(Previously Amended)* Method according to claim 17, wherein said protected part of said memory is a first part, and said memory further comprises a second part into which data can be written, where the program routine from said protected part executed by said processor upon start-up comprises checking for changes in at least a part of the data contained in said second part.

20. (original) Method according to claim 19, wherein said program routine from said protected part executed by said processor upon start-up comprises calculating a characteristic parameter for data being checked for changes, and comparing said characteristic parameter with a value stored in said second part of said memory at the time of writing said data being checked for changes into said second part of said memory.

21. (original) Method according to claim 20, wherein said characteristic parameter is a check sum.

22. (previously amended) Method according to claim 17, wherein said memory comprises a plurality of memory devices, one of which comprises said protected part, and the rest of which are arranged such that data may be written into them.

23. (cancel)

24. (original) Method according to claim 23, wherein said protected area is arranged such that the process for storing data therein comprises:

writing data into said protected part via a write line, and

sending a signal to said protected part in response to which said write line is permanently interrupted.

25. (original) Method according to claim 24, wherein said write line is a fusable link.

C1

26. (previously amended) Method according to claim 17, wherein said memory comprises a finite state machine, said finite state machine defining a state which protects said protected part from being written into.

27. (amended) Method according to claim 17, wherein said memory comprises one or more of an EEPROM, a flash memory device, and a flash memory device emulating an EEPROM.

28. (previously amended) Method according to claim 17, wherein said memory ~~comprise~~ comprises a memory chip having electrical contacts for being connected with a circuit board that are arranged such that said electrical contacts are covered by said memory chip when said memory chip is mounted on said circuit board.

29. (original) Method according to claim 28, wherein said electrical contacts are provided in a ball-grid-array.

30. (previously amended) A medium readable by a data processing device, having a program recorded thereon, where the program is to make the data processing device execute the method of claim 17.

Claims 1-2, 6-9, 11-18, 22-25, and 27-30 stand rejected under 35 U.S.C. is being clearly anticipated by U.S. patent 6,009,495 to DeRoo et al. This rejection is respectfully traversed.

DeRoo describes a particular method and hardware configuration for protecting an address range in an EEPROM. The specific technique is described in columns 87 and 88, and the hardware is shown in Figures 26 and 27. An interface 720 shown in Figure 26 is located between the host CPU and a common memory device 704. If a write access is desired, the memory address is compared in a boot block protective decoder 2050 with the predetermined address range defining the protective boot block. The boot block address range is defined by comparator input HUICFG_1.

WRITE operations are only allowed outside this selected address range. As described in column 87, lines 45-49, the HUICFG_1 bits are configured in hardware with pull up or pull down resistors. Thus, a key feature to DeRoo's memory protection system is flexibility of the boot block memory configuration.

In contrast to DeRoo, the present invention focuses more on security than on flexibility. Indeed, both independent claims 1 and 17 recite the security feature of irreversibly blocking the writing of data into the protected part of memory. Such security feature is not disclosed or suggested in DeRoo.

Claim 1 specifically recites:

> where said protected part is arranged such that a mechanism is provided such that after data is initially stored in said protected part, any subsequent writing of data into said protected part is irreversibly blocked.

Claim 17 recites the following steps:

> at least part of said memory as a protected part from which data read but which is protected against being written into, after data is initially stored said protected part, irreversibly blocking any subsequent writing of data into said protected part.

In the rejection of claims 7 and 23, the Examiner contends that DeRoo blocks "any subsequent writing of data." Reference is made to columns 99, 100, and 102. Those referenced portions of DeRoo indicate that writing to the protected address range in the memory device is prevented (in DeRoo's claim 1) using a gating mechanism (DeRoo's claim 17). But protecting a portion of memory from writing is not the same thing as irreversibly blocking any subsequent writing data into a protected part of memory as recited in amended claims 1 and 17. Indeed, DeRoo's block protection feature is not only reversible, it is optional. Column 88, line 6 states "when the blocked protection feature is enabled," clearly indicating that there are times when the block protection feature may be disabled to permit writing to that boot block. Moreover, by manipulating the values of HUICFG_1, one may readily change the memory contents of the particular area being protected. Therefore, a previously protected area of the memory can be written to. In either case, DeRoo does not describe irreversible blocking of a part of the memory. Indeed, one could remove DeRoo's EEPROM memory from the overall circuit and reprogram the blocked portion using an EEPROM reprogramming device.

Because DeRoo fails to disclose irreversible blocking of a protected portion of memory, Applicants respectfully submit that the rejection based upon DeRoo, and the subsequent obviousness rejections based upon DeRoo, are improper and should be withdrawn. The application is in condition for allowance and a notice to that effect is earnestly solicited.

Respectfully submitted,

NIXON & VANDERHYE P.C.

By: _____
John R. Lastova
Reg. No. 33,149

JRL:trb
1100 North Glebe Road, 8th Floor
Arlington, VA 22201-4714
Telephone: (703) 816-4000
Facsimile: (703) 816-4100